

(43)公開日 平成10年(1998)10月13日

A

(74)代理人 弁理士 秋田 収喜

図1

OSの入出力制御装置

表示装置

キーボード

マウス

ディスク装置

外部装置

プリンタ

被テストプログラム

操作手順記憶部

操作手順データ

操作自動実行部

テスト結果

テスト結果管理

テスト結果検出

表示装置

【特許請求の範囲】

【請求項 1】 被テストプログラムをテストするための操作手順を操作手順データとして保存しておき、その保存しておいた操作手順データを被テストプログラムに与えて被テストプログラムのテストを行うプログラムテスト支援装置において、

被テストプログラムを前記操作手順データに従って動作させるオペレーティングシステムとの間に介在し、被テストプログラムから出力される全ての情報を時系列に取得し、取得した時刻の情報と共にテスト結果情報として保存するテスト結果保存手段と、

複数回のテスト結果をテスト結果履歴として保存するテスト結果履歴保存手段と、

テスト時刻の異なる 2 つのテスト結果情報を前記テスト結果履歴保存手段から取り出して比較することにより差分を抽出する比較手段と、

抽出された差分で示される情報のみを表示する表示手段と、を備えることを特徴とするプログラムテスト支援装置。

【請求項 2】 前記テスト結果保存手段に保存する被テストプログラムから出力される全ての情報を出力命令形式の文字列情報として前記テスト結果保存手段に保存する出力命令文字列変換手段を備えることを特徴とする請求項 1 記載のプログラムテスト支援装置。

【発明の詳細な説明】**【0001】**

【発明の属する技術分野】 本発明は、画面操作を行いながら新規に開発したプログラムのテストの支援を行うプログラムテスト支援装置に関するものである。

【0002】

【従来の技術】 従来、この種のプログラムテスト支援装置として、例えば特開平 5 - 2 7 4 1 8 6 号公報に開示されたものが知られている。

【0003】 従来の技術の特開平 5 - 2 7 4 1 8 6 号を例に挙げて説明すると、この特開平 5 - 2 7 4 1 8 6 号（発明の名称：入力データ処理装置）では、オペレータの操作指示の情報をロギングしておき、その情報を再度実行することでプログラムのテストを自動化している。テストの結果については、比較の必要な画面については、テスト操作の中で画面のログをとるキーを入力し、指示された時点での画面のハードコピーをファイル化して 2 度目のテストの結果と比べるようになっている。

【0004】

【発明が解決しようとする課題】 しかしながら、上記の従来技術にあつては、次のような問題点がある。

【0005】 (1) オペレータの指示のない画面については画面のログが取得されない。したがって、予め変更が予測できる画面についてはログを取得しておく操作を指示できるが、テスト対象のプログラムの変更に伴い、予測しない画面の変更が起きた場合などには対応するこ

とができない。

【0006】 (2) 特開平 5 - 2 7 4 1 8 6 号では、画面の中でブリンクしているフィールドに対して、ブリンクを止めてから画面のハードコピーを取得することが特徴となっているが、画面のエラーメッセージのようにブリンクすることが仕様として要求されているフィールドに対して、画面の状態を変更してしまうことにより、ブリンクしていたのか、していなかったのかの確認ができなくなってしまう。

10 【0007】 (3) 画面のログを取得する行為をテストのオペレーションと同一レベルで扱ってしまうため、画面の出現順序や画面の追加や削除といった画面の流れの確認はできない。

【0008】 (4) 画面のログを取得するために特別なキーが必要であり、現在のように表示装置上に複数のプログラムの実行を示す画面が表示されている場合、目的のプログラムの出力する画面の情報だけを取得する機能が必要であり、目的以外のプログラムの出力する画面に依存しないテストを実行する必要がある。また、特開平 5 - 2 7 4 1 8 6 号では、すべての画面情報をログに取得してしまうため、同時に実行しているプログラムがある場合、画面の情報が異なる可能性が高くなり、正常にテストを実施するには、表示装置全体を一つのプログラムの出力する画面とする必要がある。

【0009】 (5) 出力のタイミングが記録できていないため、性能の劣化が起きていることを比較することができない。

30 【0010】 (6) テスト対象のプログラムによっては、画面出力だけでなく、ファイル出力や外部プログラムへのメッセージ出力など、出力するデータが多種あるが、上記従来技術はすべての出力情報に対して配慮していない。

【0011】 本発明の目的は、画面やファイル、外部装置等のテスト対象のプログラムから出力される全ての情報を時系列に残し、画面の遷移を容易に確認でき、また予測しない画面の変更に対しても容易に対処でき、さらには性能の劣化を評価することができるプログラムテスト支援装置を提供することにある。

【0012】

40 【課題を解決するための手段】 上記目的を達成するために、本発明は、被テストプログラムをテストするための操作手順を操作手順データとして保存しておき、その保存しておいた操作手順データを被テストプログラムに与えて被テストプログラムのテストを行うプログラムテスト支援装置において、被テストプログラムを前記操作手順データに従って動作させるオペレーティングシステムとの間に介在し、被テストプログラムから画面、ファイル、外部装置に出力される全ての情報を時系列に取得し、取得した時刻の情報と共にテスト結果情報として保存するテスト結果保存手段と、複数回のテスト結果をテ

スト結果履歴として保存するテスト結果履歴保存手段と、テスト時刻の異なる2つのテスト結果情報を前記テスト結果履歴保存手段から取り出して比較することにより差分を抽出する比較手段と、抽出された差分で示される情報のみを表示する表示手段とを備えることを特徴とする。

【0013】また、前記テスト結果保存手段に保存される保存被テストプログラムから画面、ファイル、外部装置に出力される全ての情報を出力命令形式の文字列情報として前記テスト結果保存手段に保存する出力命令文字列変換手段を備えることを特徴とする。

【0014】

【発明の実施の形態】以下、本発明を図面に示す実施形態に基づいて詳細に説明する。

【0015】図1は、本発明を適用したプログラムテスト支援装置の実施形態を示すブロック図である。

【0016】本実施形態のプログラムテスト支援装置10は、操作手順を記録した操作手順データファイル11、テスト実行時に出力された情報を格納したテスト結果ファイル12、何度かテストを繰り返すときに過去のテスト結果をテスト結果履歴として保存しているテスト結果履歴ファイル13、オペレータの操作手順を記録する操作手順記録部20、操作を操作手順データファイル11中の操作手順に基づき自動で実行する操作自動実行部30、被テストプログラム70が出力する情報を保存するテスト結果保存部40、テストを繰り返し実行する場合に過去のテスト結果を保存したり、比較のために過去のテスト結果を取り出したりするテスト結果管理部50、テストして得られた結果を過去の結果と比較するテスト結果検証部60で構成されている。

【0017】テストの対象となる被テストプログラム70が動作する計算機システムの環境は、オペレーティングシステム（以下、OS）の入出力制御部80、ディスプレイなどの表示装置81、文字やコマンド等を入力するためのキーボードなどの入力装置82、マウスなどのポインティング装置83、ファイルなどを格納するディスクなどの外部記憶装置84、外部ネットワークやシリアルインタフェースなどの外部装置85、プリンタ装置86を備えている。なお、テスト結果検証部60の検証結果は、表示装置81に表示される。

【0018】まず、この実施形態の動作の概要について説明する。

【0019】被テストプログラム70に対するオペレータの入力情報をOSの入出力制御部80から取得し、操作手順記録部20が操作手順データファイル11に記録する。被テストプログラム70が出力する情報は、OSの入出力制御部80に命令を渡す前に、テスト結果保存部40がテスト結果ファイル12に記録する。

【0020】2回目のテストを始める前に、テスト結果管理部50を使って1回目のテスト結果をテスト結果フ

ァイル12から取得し、テスト結果履歴ファイル13に転送する。

【0021】以降のテストは、1回目のテストで操作手順データファイル11に記録した操作手順データに基づき操作自動実行部30がOSの入出力制御部80に命令を送る。OSの入出力制御部80は受け取った命令を、キーボード等の入力装置82やポインティング装置83からの指示と同様に処理し、被テストプログラム70に渡す。

【0022】被テストプログラム70が出力する情報を1回目と同じようにテスト結果保存部40がテスト結果ファイル12に記録する。1回目に記録したテスト結果Aと、2回目のテスト結果Bとを比較するために、1回目のテスト結果Aをテスト結果管理部50を使ってテスト結果履歴ファイル13から取り出す。

【0023】テスト結果検証部60は、取り出した1回目のテスト結果Aと2回目のテスト結果Bを使い、テスト結果の違いを表示装置81に出力する。

【0024】図2は、テスト結果保存部40の機能の詳細を示す機能ブロック図である。図中の41は画面に出力する命令をテスト結果に記録する画面出力機能、42は外部記憶装置84のファイルに出力する命令をテスト結果に記録するファイル出力機能、43はプリンタ装置86に出力する命令を記録するプリンタ出力機能、44は外部装置85に出力する命令を記録する外部装置出力機能である。

【0025】前記画面出力機能41において、411は画面出力機能41の内部の詳細機能である出力命令を文字列に変換する出力命令文字列変換機能、412は変換した文字列を格納する文字列格納エリア、413は文字列をテスト結果ファイル12に出力するファイル出力機能である。

【0026】一方、OSの入出力制御部80において、801は表示装置81に表示画面内容を直接出力する画面出力機能、802は同じく外部記憶装置84にファイルデータなどを直接出力するファイル出力機能、803は同じくプリンタ装置86にプリントデータを直接出力するプリンタ出力機能、804は同じく外部装置85に外部出力データを直接出力する外部装置出力機能である。

【0027】図3は、テスト結果を記録する方法を示すフローチャートである。図2と図3を参照し、画面出力の情報をテスト結果ファイル12に記録する方法を説明する。

【0028】被テストプログラム70は、表示装置81に表示する場合は、OSの入出力制御部80の画面出力機能801を使用する。本実施形態では、被テストプログラム70が直接、画面出力機能801を使用するのではなく、テスト結果保存部40の画面出力機能41を使用する。

【0029】まずはじめに、被テストプログラム70は、テスト結果保存部40の画面出力機能41に画面出力命令を送る(図3のステップ301)。画面出力機能41は画面出力命令を受け取ると(ステップ302)、その受け取った命令を出力命令文字列変換機能411を使い文字列に変換し、文字列エリア412にセットする(ステップ303)。

【0030】文字列に変換する理由は、テスト結果の検証を簡単にするためである。次に、変換した文字列に時刻、画面名などの必要情報を付加した後(ステップ304)、文字列エリア412の内容をファイル出力機能413によりテスト結果ファイル12に出力する(ステップ305)。

【0031】これにより、被テストプログラム70が画面に出力する情報は、出力命令形式の文字列に変換され、かつ時刻、画面名などの情報が付加されてテスト結果ファイル12に時系列で記録される。出力命令形式の文字列に変換してテスト結果ファイル12に記録することにより、テスト結果ファイル12の記憶容量をドットイメージで記録する場合に比べて小さくすることができる。また、画面の情報をイメージデータで取得するとブリンクしている項目などを正しく記録できないが、画面上に文字や線を描画する機能を実行する度に描画情報を出力命令形式の文字列情報(スクリプト言語)で記録することにより、描画された文字や線の属性を正確な情報として記録することができる。

【0032】図4は、テスト結果管理部50とテスト結果検証部60の詳細構成を示す機能ブロック図である。図中の12aはテスト結果履歴ファイル13中に格納される最新のテスト結果であり、12bは過去のテスト結果である。501は最新のテスト結果12aからテスト実行の日時を取得するモジュール、502はテスト結果12aをテスト結果履歴ファイル13に格納するテスト履歴格納モジュールである。

【0033】503はテスト結果履歴ファイル13の中から指定された過去のテスト結果12bを取り出すテスト履歴取り出しモジュール、504は過去のテスト結果12bとして必要な日時の情報を入力装置82から取得する日時指定モジュールである。

【0034】一方、テスト結果検証部60において、601は最新のテスト結果12aと過去のテスト結果12bを入力して、テスト結果の差分を出力するテスト結果比較部、602はテスト結果の差分情報を格納するテスト結果差分情報ファイル、603は求めた差分情報を表示装置81に出力する差分表示部である。この差分表示部603の内部は、テスト結果差分情報ファイル602に格納された差分情報から表示装置81に表示するための命令語を生成する画面出力命令生成モジュール604と、生成した命令文を格納する命令文格納エリア605、この命令文格納エリア605に格納した命令文を○

Sの入出力制御部80を使用して表示装置81に出力するOSの画面出力命令実行モジュール606から構成されている。

【0035】図5(a)、(b)は、テスト結果ファイル12の内容を示す図である。同図(a)はテスト結果ファイル12に格納されている内容の詳細、同図(b)は表示装置81に実際に出力された画面の例である。1201は画面に出力した内容を表す識別子、この場合はボタンを表している。1202はボタンに表示する文字列、1203はボタンの形式を表すコード、1204はボタンを表示する位置をx、y座標で表す。1205はボタンの大きさをx、y方向に表す。位置と大きさの単位はドットである。1206はこのボタンがどの画面に出力されたのかを識別するための識別子であり、この場合は、画面の名称である「確認」の文字列が使われている。1207は実際に表示装置81に出力された画面、1208は画面の名称が表示されている部分、図中の数字は起点1213からの距離をドット単位で表している。1209はテキスト入力フィールドを表す記録結果、1210はボタンを表す記録結果、1211は実際の画面に出力されたテキスト入力フィールド、1212は実際の画面に出力されたボタンを表す。

【0036】図示のように、ボタン1212は、起点1213を基準にして、ボタンを表示する位置座標1204で指定されるドット位置に、ボタンの大きさを表すx、y方向の大きさの情報1205で指定される大きさ表示される。テキスト入力フィールド1211についても同様に、記録結果1209の中のx、y座標の値に従って表示される。

【0037】図6は、複数回テストを実行する場合のテスト結果履歴をテスト結果履歴ファイル13に格納する手順を示したフローチャートである。図4を参照して説明する。

【0038】まず、最新のテスト結果12aからテストした日時を日時取得モジュール501で取得する(ステップ6001)。次に、ファイルの属性として、取得したテスト実施日時を付加する(ステップ6002)。次に、テスト履歴格納モジュール502が、最新のテスト結果12aをテスト結果履歴ファイル13に格納する(ステップ6003)。

【0039】図7および図8は、テスト結果検証部60の動作を説明する図である。図7において、701は1回目のテスト結果ファイル12の内容であり、図中の2重線は2回目のテストで値が変わった内容を示す。705はテキスト入力フィールドであり、706は文字列がOKと表示されるボタン、707は文字列にキャンセルと表示されるボタン、702は2回目のテスト結果ファイル12の内容であり、図中の2重線は1回目と値が変わった内容を示す。708はテキスト入力フィールド、709は文字列に送信と表示されるボタンであり、70

5～710には出力日時が付加されている。710は文字列にキャンセルと表示されるボタン、703は1回目と2回目のテスト結果の差分情報602、711は1回目と表示位置が異なったテキスト入力フィールド、712は1回目と文字列と表示位置が異なったボタンである。701の先頭行の出力日時と最終行の出力日時の時間間隔と702の先頭行の出力日時と最終行の出力日時の時間間隔を比較することでプログラム全体の動作時間の差分が求められる。

【0040】また、705の出力日時と706の出力日時の時間間隔と707の出力日時と708の出力日時の時間間隔を比較することにより、各部分でのプログラムの動作時間の差分が得られる。

【0041】次に、図8において、704はテスト結果差分情報703を画面に表示した結果、713は表示位置が変更になったテキスト入力フィールド、714はボタンの文字列と表示位置が変わったボタンを示す。

【0042】以上のように本実施形態によれば、テスト結果ファイル12の内容を単純にテキストレベルで比較するだけで、被テストプログラム70における画面操作が変更になった部分を抽出し、表示装置81に表示し、変更部分を確認することができる。従って、テスト対象のプログラムの変更に伴い、予測しない画面の変更が起きた場合に、容易に対処することができる。また、出力日時の情報から出力時間間隔を比較することで、ファイルやデータの出力単位で性能の劣化を検出することができる。

【0043】さらに、出力命令形式の文字列に変換してテスト結果ファイル12に記録することにより、テスト結果ファイル12の記憶容量をドットイメージで記録する場合に比べて小さくすることができる。また、画面の情報をイメージデータで取得するとブリンクしている項目などを正しく記録できないが、画面上に文字や線を描画する機能を実行する度に描画情報を出力命令形式の文字列情報（スクリプト言語）で記録することにより、描画された文字や線の属性を正確な情報として記録することができる。

【0044】さらに、プログラムテスト支援装置10をOSの入出力制御部80との間に介在させ、被テストプログラムが出力する全ての情報を取得しているため、画面上に複数のプログラムの実行画面が表示されているマルチタスク環境においても、目的とする1つの被テストプログラムの出力情報のみを取得し、テスト結果を確認することができる。

【0045】

【発明の効果】以上の説明から明らかなように、本発明によれば、画面やファイル、外部装置等のテスト対象のプログラムから出力される全ての情報を時系列に残し、画面の遷移を容易に確認でき、また予測しない画面の変

更に対しても容易に対処でき、さらには性能の劣化を評価することができる。

【0046】また、プログラムテスト支援装置をOSの入出力装置との間に介在させ、被テストプログラムが出力する全ての情報を取得しているため、画面上に複数のプログラムの実行画面が表示されているマルチタスク環境においても、目的とする1つの被テストプログラムの出力情報のみを取得し、テスト結果を確認することができる。

【0047】さらに、出力命令形式の文字列に変換してテスト結果ファイルに記録することにより、テスト結果ファイルの記憶容量をドットイメージで記録する場合に比べて小さくすることができる。また、画面の情報をイメージデータで取得するとブリンクしている項目などを正しく記録できないが、画面上に文字や線を描画する機能を実行する度に描画情報を出力命令形式の文字列情報（スクリプト言語）で記録することにより、描画された文字や線の属性を正確な情報として記録することができる。

【図面の簡単な説明】

【図1】本発明を適用したプログラムテスト支援装置の実施形態を示すブロック図である。

【図2】テスト結果保存装置の機能の詳細を示す機能ブロック図である。

【図3】テスト結果を記録する方法を示すフローチャートである。

【図4】テスト結果管理装置とテスト結果検証装置の詳細構成を示す機能ブロック図である。

【図5】テスト結果ファイルの内容を示す図である。

【図6】複数回テストを実行する場合のテスト結果履歴をテスト結果履歴ファイルに格納する手順を示したフローチャートである。

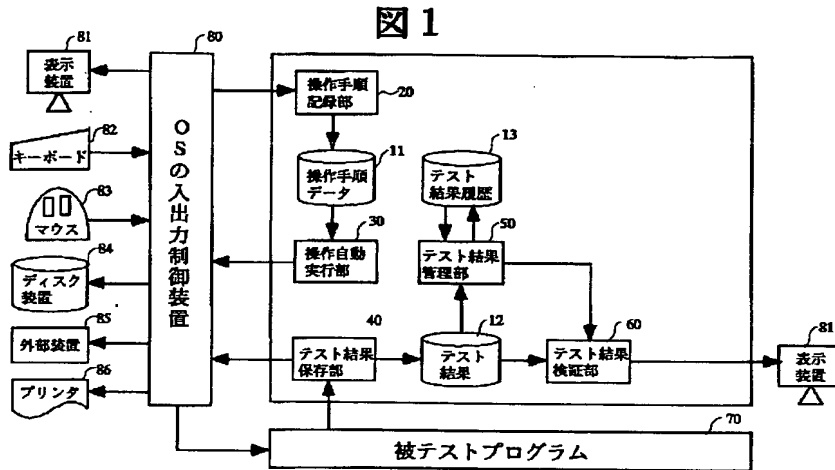
【図7】テスト結果検証装置の動作を説明する図である。

【図8】テスト結果差分情報を画面に表示した結果を示す図である。

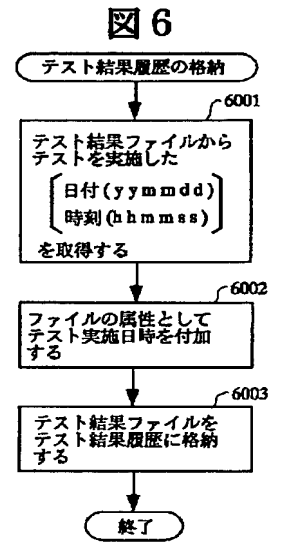
【符号の説明】

10…プログラムテスト支援装置、11…操作手順データファイル、12…テスト結果ファイル、13…テスト結果履歴ファイル、20…操作手順記録部、30…操作自動実行部、40…テスト結果保存部、41…画面出力機能、42…ファイル出力機能、43…プリンタ出力機能、44…外部装置出力機能、50…テスト結果管理装置、60…テスト結果検証部、70…被テストプログラム、80…OSの入出力制御部、81…表示装置、82…入力装置、84…外部記憶装置、85…外部装置、86…プリンタ装置、411…出力命令文字列変換機能、601…テスト結果比較部、602…テスト結果差分情報ファイル、603…差分表示部。

【図 1】

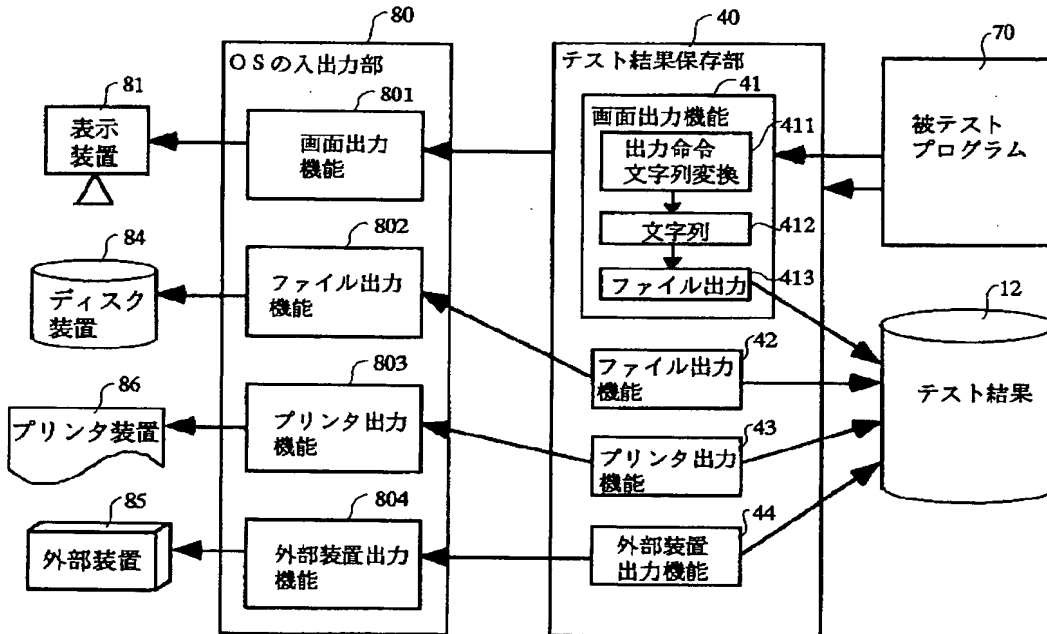


【図 6】

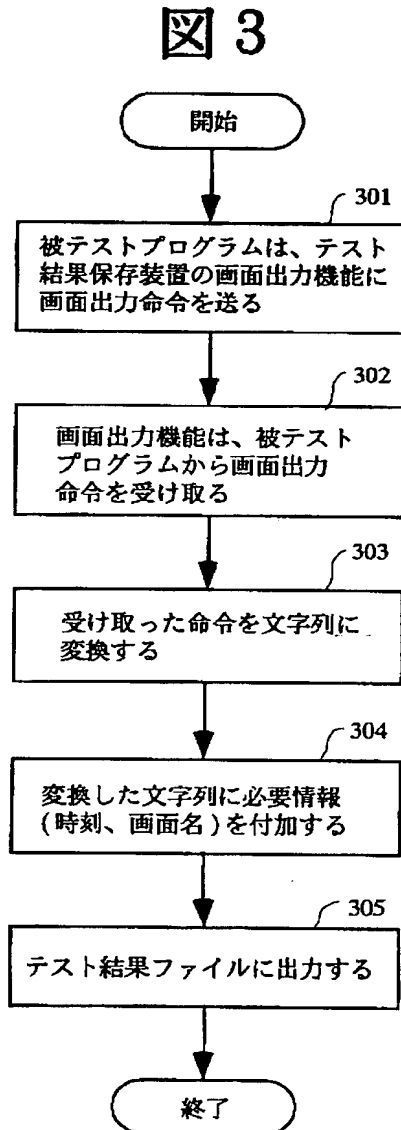


【図 2】

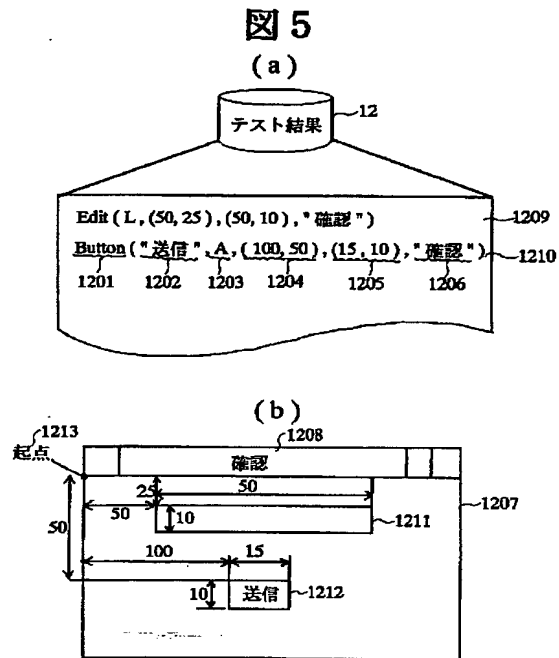
図 2



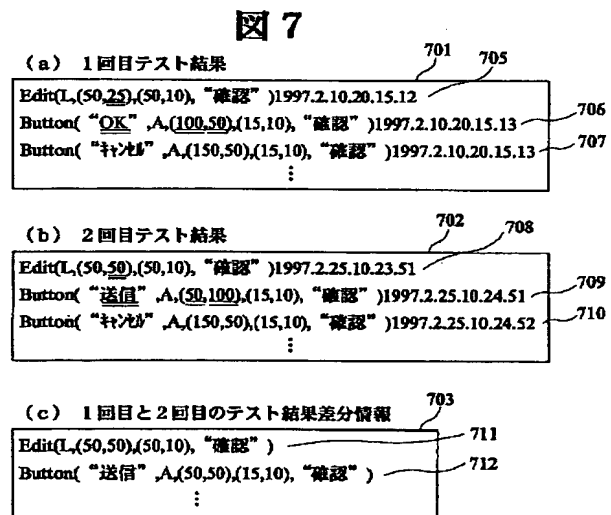
【図 3】



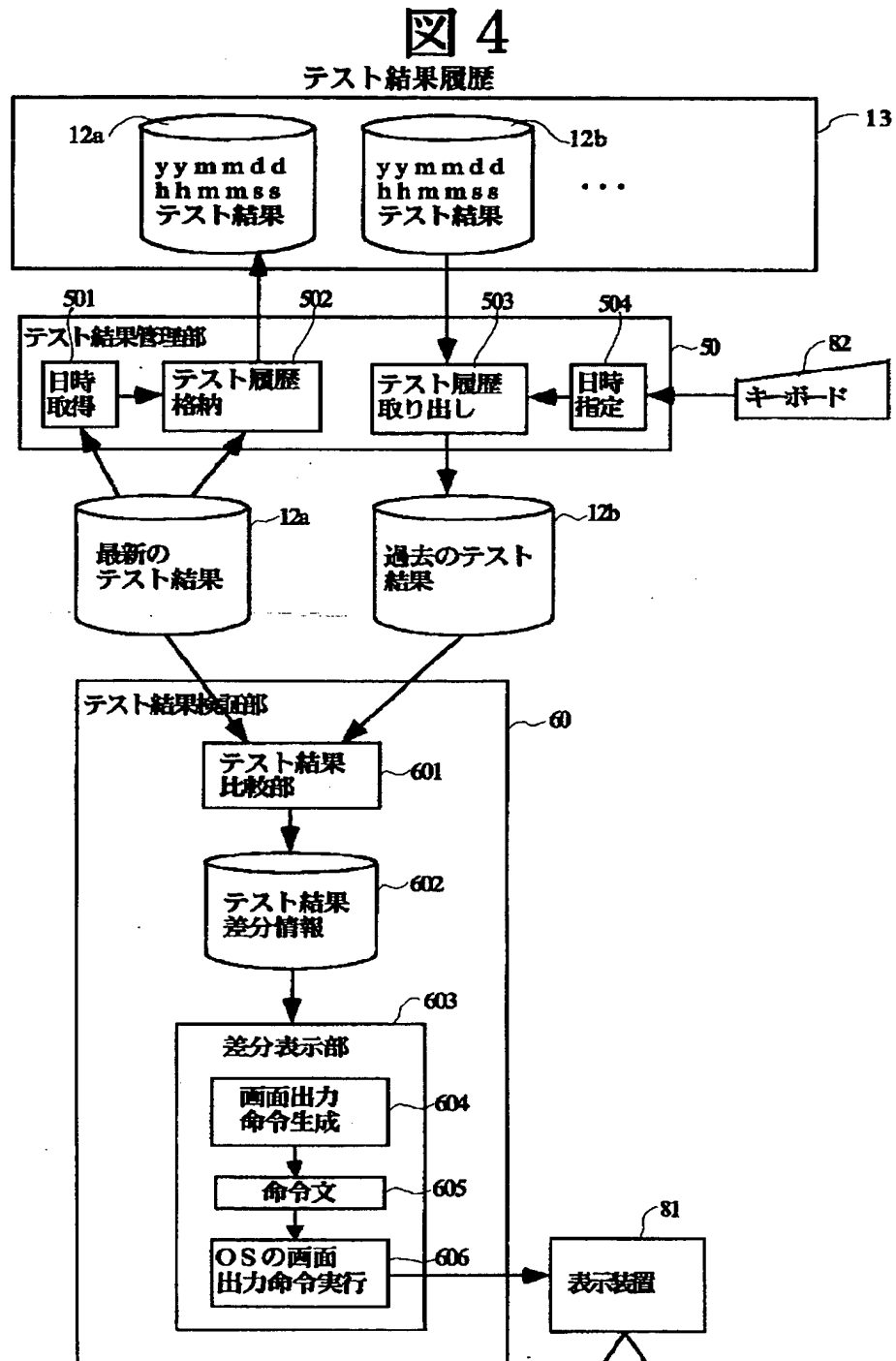
【図 5】



【図 7】

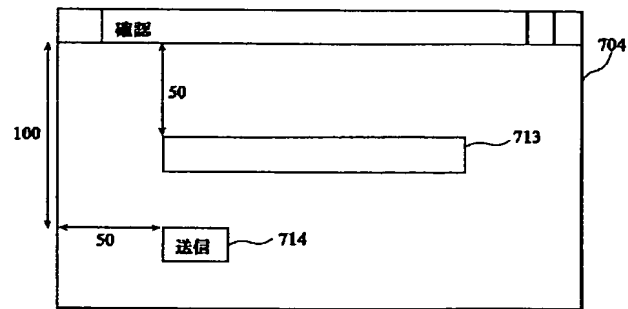


【図 4】



【図 8】

図 8



Abstract

BACKGROUND OF THE INVENTION

General program test support system conducts the automatic test by taking the user's operation log and by re-executing the log.

The test result, which the screen comparison is necessary, will be shown by:

First by inputting a key to log the screen operation and file the hardcopy of the indicated screen, and then compare with the latter test result.

Issues appeared as follows:

- (1) The screen log would not be kept without the operator's command. Therefore, the screen predicted to be modified can be indicated to keep its log, however, the log can not be taken under unpredictable screen modification due to the change of test program.
- (2) A hardcopy of the screen will be taken after its field has stopped blinking. However, if the screen is required to blink at specific point due to the specification, just like a error message, it is difficult to confirm whether a blink occurred or not by means of the change to the screen status.
- (3) Since the system considers the operation of logging the screen and testing to be the same operation, it is unable to confirm the screen flow such as screen appearance order and/or added or deleted screen.
- (4) If the system requires a specific key to obtain the screen log and if several screens which indicates the program execution appears likewise general system,
 - The function to get only necessary screen information from the indicated program is required.
 - The test to check whether the screen is independent from the unintended screen output should be conducted.

In addition, as all the screen information is kept into the log, if there are several programs being executed, there are much possibilities that the screen information might not match, thus, the screen must be output from one single program in order to conduct the reliable test.

- (5) Since the records of output timing do not exist, the system is unable to compare and determine the program's quality inferiority.
- (6) General system does not correspond to various output data of the testing program, such data as screen output, file outputs and output messages to external programs.

AIM OF THE INVENTION

To provide a program testing system, which is able to keep all data record, such as screens/ files/external devices derived from the program needed for the test, to check the screen flow easily, to deal with unexpected screen changes, and also to evaluate the program's quality inferiority.

SUMMARY OF THE INVENTION

(1)

To validate the tested program, the operation method can be recorded as operational procedure data. The program test support system will provide the procedure data for the tested program in order to run the validation. By placing the tested program between the operating system, which operates according to the operation procedure data, the program test support system will collect all data from the tested program in time-order. The program test system is consisted of:

- Test result storing method, to store the test result data which includes data stored time
- Test result history storing method, to store several test results as the test result history
- Comparison method, to take out two separate test results, which its time differs, from the test result history storing method and extract the difference to compare
- Data listing method, showing only the difference data extracted.

(2)

In addition, this system provides the string change method of output instruction, to store all data outputted from the tested program which is recorded to the test result storing method and convert into string data in output instruction format, and then storing into the test data storing method mentioned above.

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☒ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.